IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

DJ

2024-02-02

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- Methodologies used in the project include:
- Data collection from the SpaceX API
- Web scraping using the Python BeautifulSoup library
- Conversion of the JSON file to a Pandas dataframe
- Exploratory data analysis using graphs and SQL queries
- Creation of interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using classification models

- Summary of all results
- Launches were much more likely to be successful in later years
- Landing success rate differs by orbit type
- Further analysis could investigate the statistical significance of our results. But the classifier models seem to be a good indicator, although the sample size is low.

Introduction

Project Background and Context

- Overview
  - SpaceX is a company launching rockets with comparable low cost
  - Key factor for cost effectiveness is reusability of the first stage
- Goal
  - launch our own rocket program with competitive prices to SpaceX Falcon 9 rockets
- Tasks
  - Identify key factors for landing the first stage successfully
  - Deduct any further insights that enables us to launch our own rockets with low cost

Section 1

Methodology

Methodology

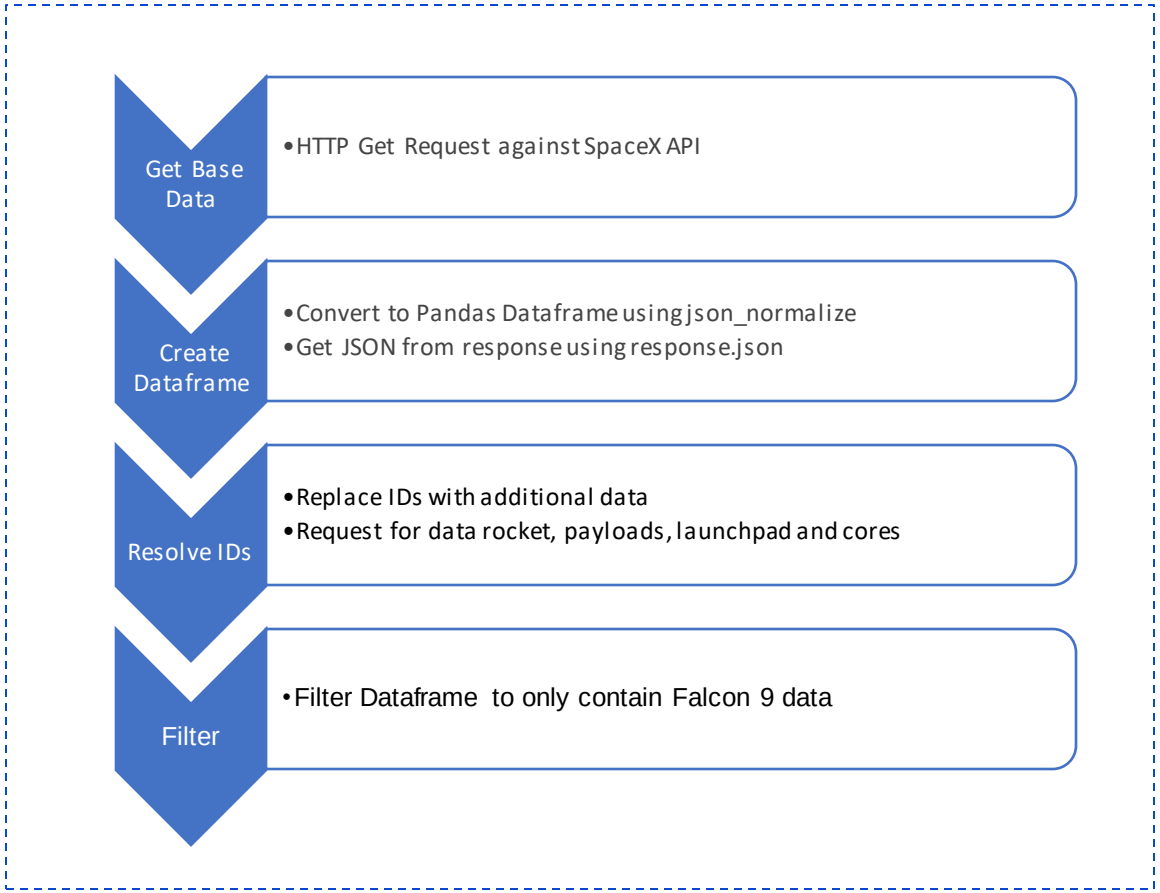
- Executive Summary
- **Data collection** methodology:
  - SpaceX data was collected from the **public SpaceX API**
  - Launch data scraped from **Wikipedia** BeautifulSoup
- Perform **data wrangling**
  - Exploratory data
- Perform **exploratory data analysis (EDA)** using visualization and SQL
- Perform **interactive visual analytics** using Folium and Plotly Dash
- Perform **predictive analysis** using classification models
  - following **classification models** were evaluated: logistic regression, decision tree classifier, K-nearest neighbor and support vector machine

Data Collection

- Data collected using **HTTP Get** against the **SpaceX API** and **Wikipedia** Page
- Multiple Endpoints of the SpaceX API and Wikipedia Data merged into a **Pandas DataFrame** using **json_normalize()**
- Overview of the additional data:
  - For rocket - booster name
  - For launchpad - name of the launch site, coordinates
  - For payload - mass of the payload, the orbit
  - For cores - type and outcome of landing, technical details

Data Collection – SpaceX API

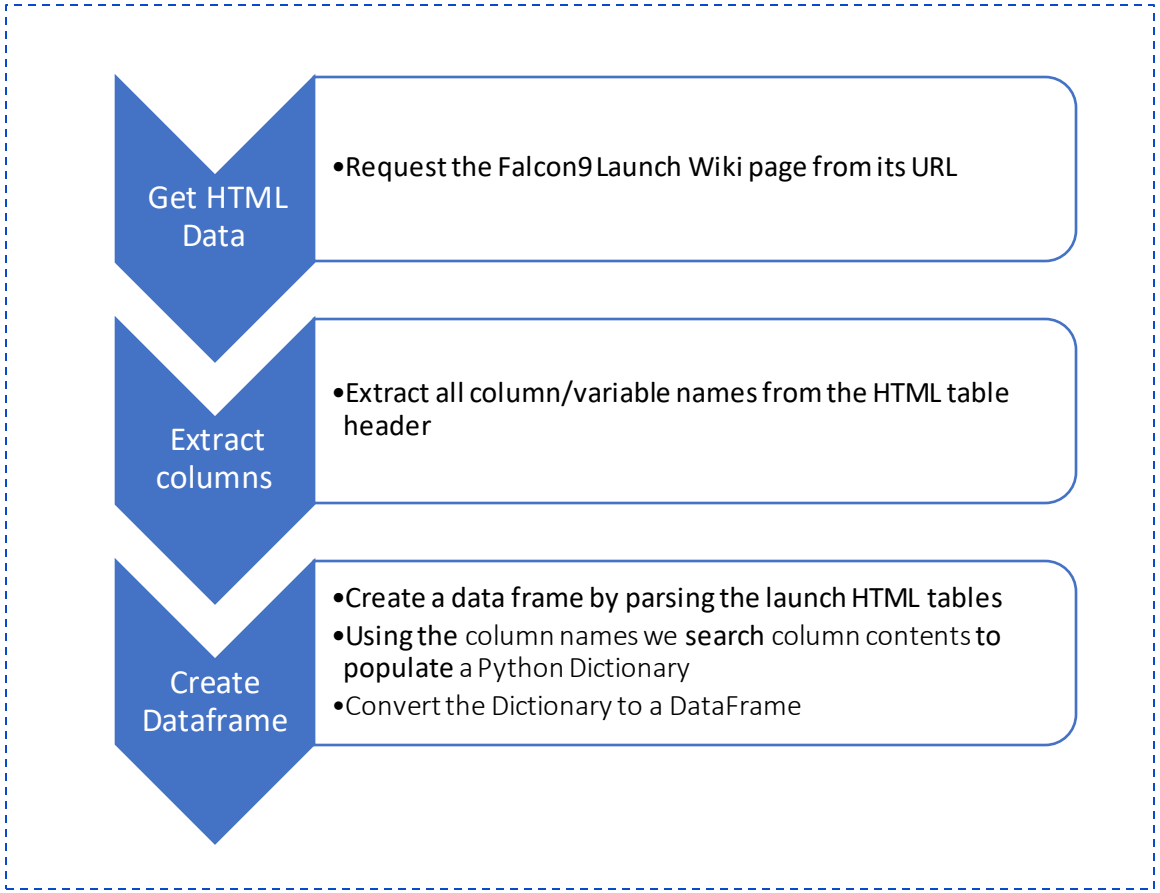
- Data from the SpaceX API has been collected via HTTP request and converted into a Pandas Dataframe
- For the final Dataset we resolved the IDs of some key columns with actual corresponding data
- Finally data was filtered for Falcon 9 rocket



Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/53bed66c98d1388f82f031b8284bb0b53997b6f0/1-1a-jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

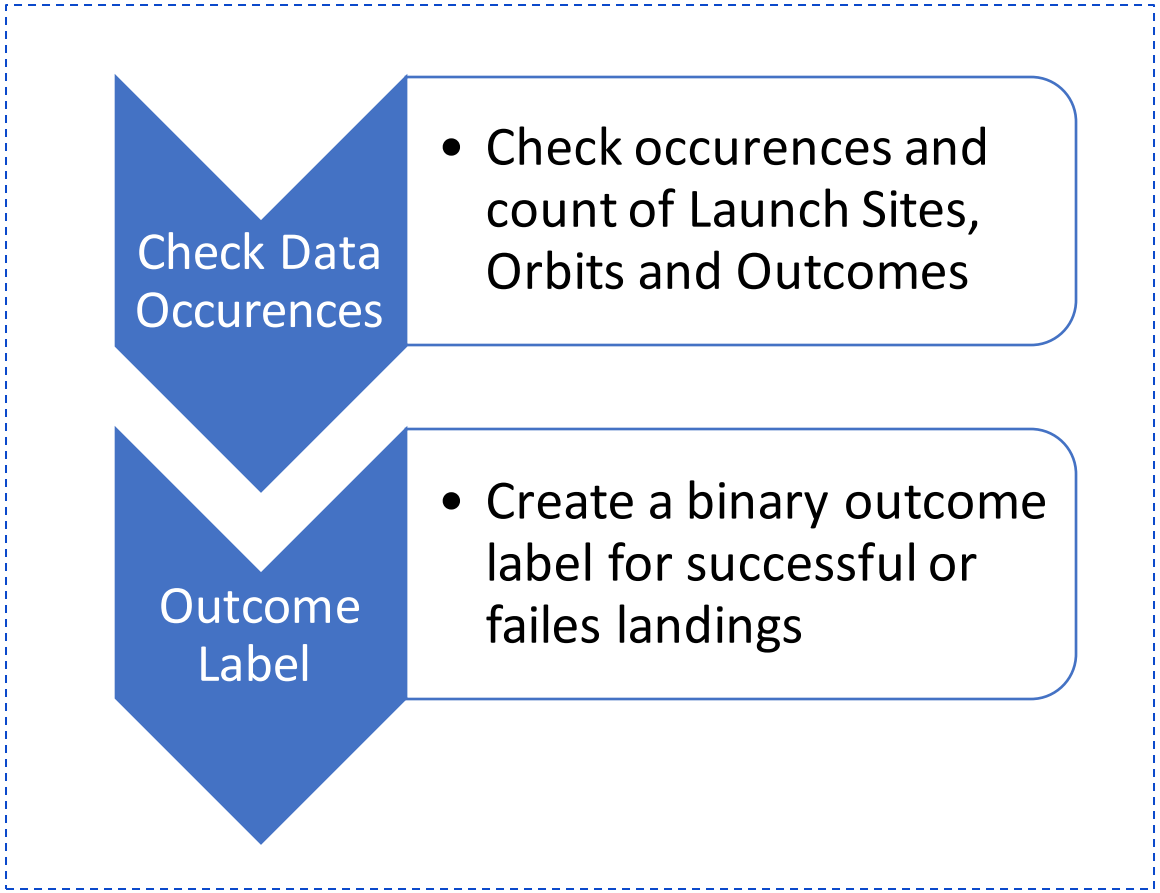
- Launch Data was aquired via Web Scraping from the Wikipedia Page of Falcon 9
- Data Extraction has been done using BeautifulSoup Package



Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/53bed66c98d1388f82f031b8284bb0b53997b6f0/1-1b-jupyter-labs-webscraping.ipynb

Data Wrangling

- Calculated **launches per site** and orbits using value_counts() method
- Determined **landing outcomes frequency** using value_counts()
- Convert landing_class into binary label column (0 or 1) for further evaluation
- Calculated the percentages for successful outcome



Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/53bed66c98d1388f82f031b8284bb0b53997b6f0/1-2-labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- Created several plots to **find correlations** and **good features** for training our **model**
- **Plots** include:
  - Relationships between **Flight Number, Payload Mass and Launch Site** (three different Scatter Plots)
  - **Success rate** of each **Orbit Type** (Bar Plot)
  - Relationship between **Flight Number** and **Orbit Type** as well as **Payload Mass** and **Orbit Type** (two Scatter Plots)
  - **Yearly trend** of Success Rate (Line Plot with confidence interval)

Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/8f2521054a877aba23cb8470ca11098c74ff1479/2-2-jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- SQL queries were executed to show the following:
  - **List of launch sites** with example data
  - Totals and averages of **payload mass**
  - Date of the **first successful landing** on ground pad
  - **Names of boosters** for specific restrictions
  - Total number of **successful and failed mission** outcomes
  - Explorations for **specific time frames**

Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/8aae01990e2083ccd836ec13caa7079d46bbe5ef/2-1-jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- The following **objects** where added using **Folium**:
  - **Launch Sites** (Circle with Popup labels) and their Name Labels (Marker)
  - **Landings markers** at the specific Launch Sites (MarkerCluster)
  - **Distance calculations** to specific infrastructure (Highways, Railroads, Cities and Coastline) near Launch Sites (Marker and PolyLine)
- The goal was to:
  - Use Folium Maps for **interactive exploration** of the surroundings of Launch Sites
  - See at a glance where the **launch sites are located**
  - **Extract patterns** based on the distance of **surroundings** to determine, where a good launch site location would be

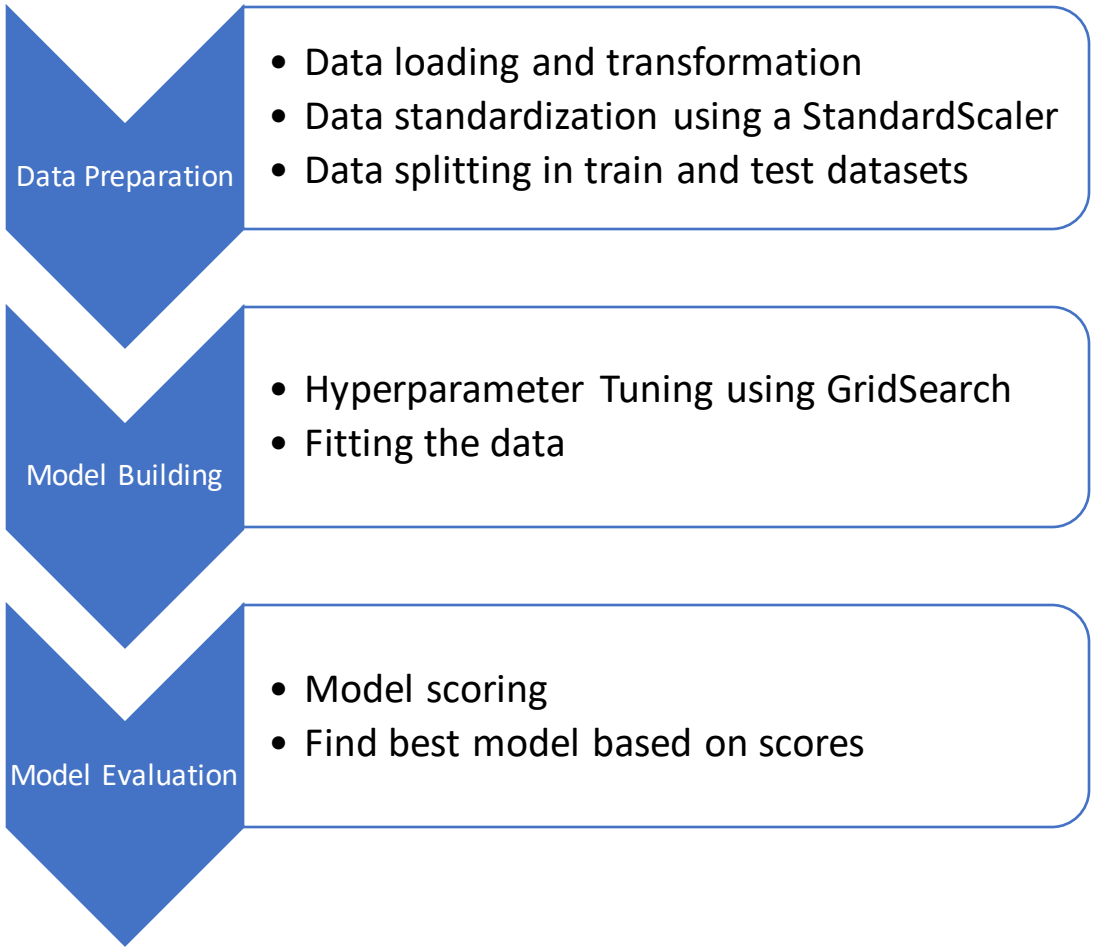
Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/3daf841912bb86d54fcf7c43388910755dcd4dc1/3-1-lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

- In the **Dash** interactive dashboard we displayed several **Plotly Plots** to:
  - Get an overall view of successful launches
  - Be able to drill into the data for specific sites
- **Pie chart** with number auf **successful launches** per site shows:
  - Distribution of successful launches between sites
  - Percentage of successful launches at that site (if filtered)
- **Scatter Plot payload** and **success** of launches to:
  - See if there is a correlation between the two values
  - There are any payload ranges that are particularly successful/unsuccessful
- Additional **Bar Chart** with the **successful launches** by **booster version** and **launch site** to:
  - See if there is a booster version that is mostly successful

Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/cb2ff1b6825d9de9f53f920b26903e1c6c7014f6/3-2-spacex_dash_app.py

Predictive Analysis (Classification)



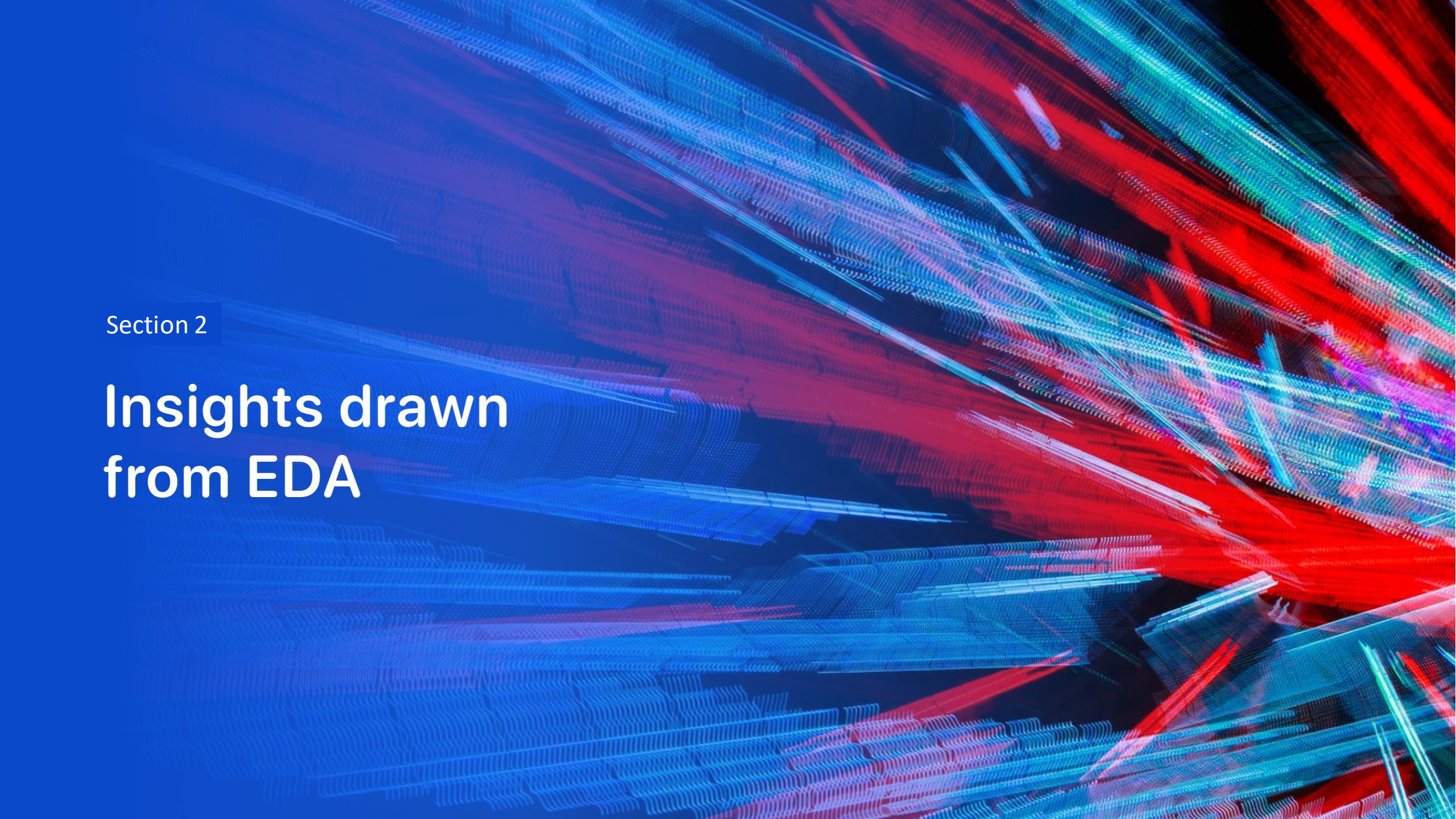
Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/2980b14fa149d8906f4aa518385a1eba3d45e404/4-SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

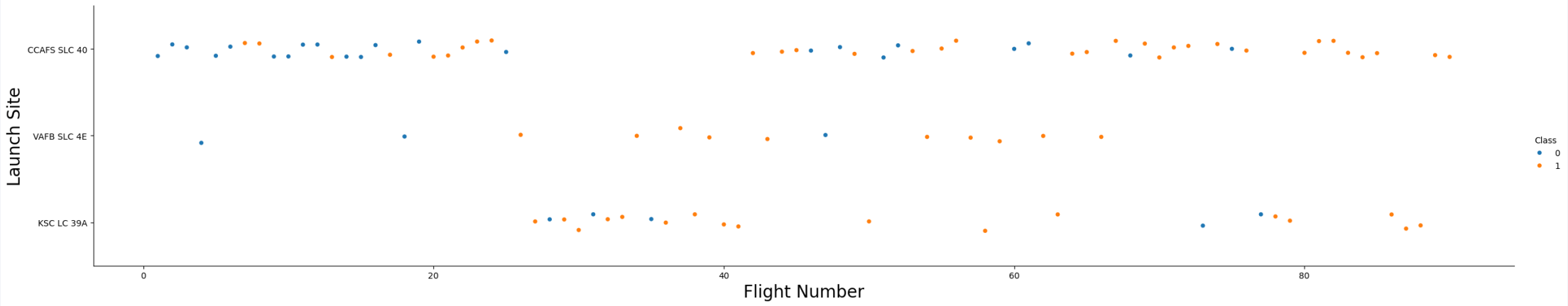
Section 2

Insights drawn from EDA



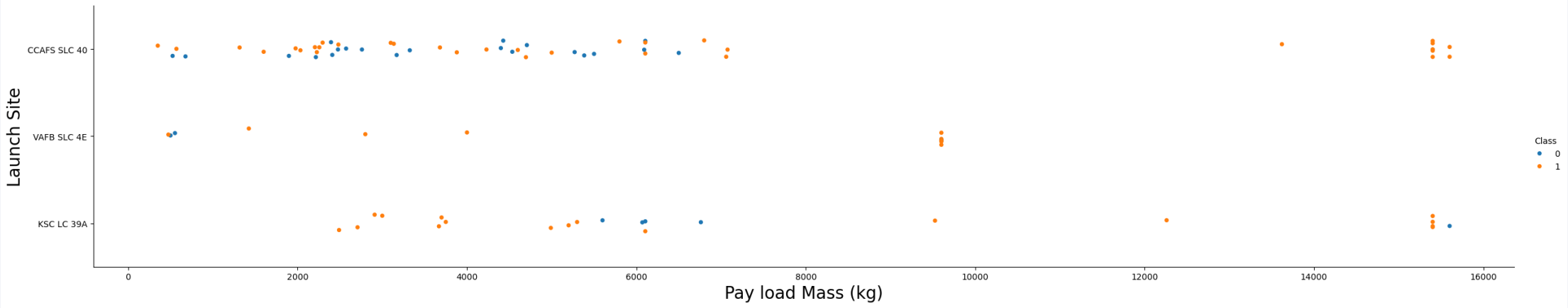
Flight Number vs. Launch Site

- Plotted Flight Number vs. Launch Site as a Scatter Plot:
  - Flights tend to launch in batches from launch sites
  - Correlation between the two seems to be small
  - Generally later flights seem to be more successful



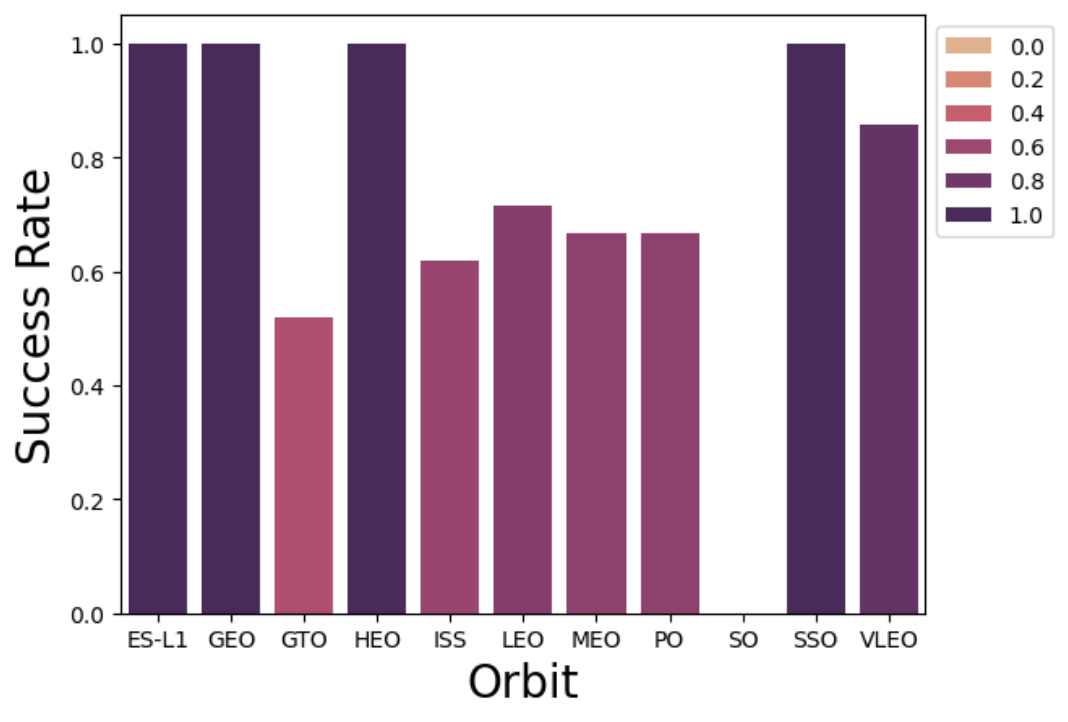
Payload vs. Launch Site

- Plotted Payload vs. Launch Site as a Scatter Plot:
  - Different launch sites have different mass profiles (VAFB not higher than 10000)
  - More Flights with lower payload (<6000) than higher payload (>6000)
  - High payload launches seem to be mostly successful



Success Rate vs. Orbit Type

- Plotted Success Rate vs. Orbit Type as a Bar Plot:

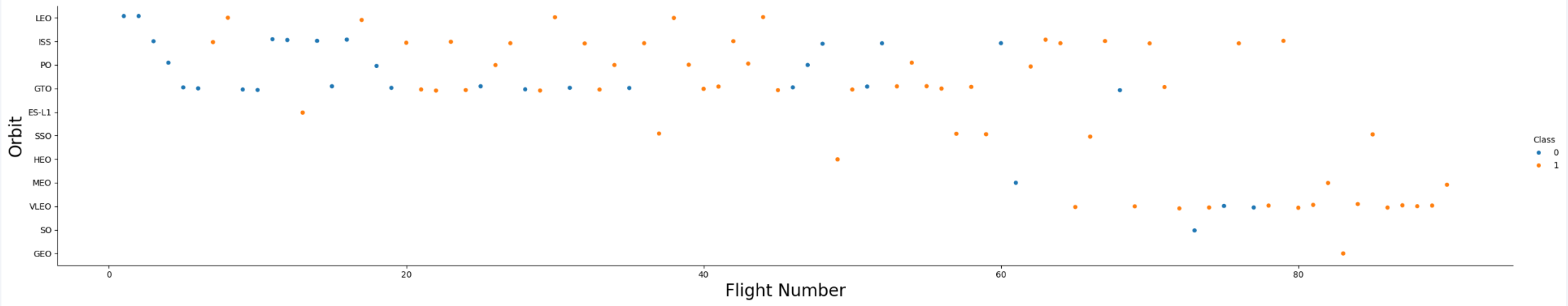


- Orbit and success rate seem to be correlated
- Orbits with a higher success chance include
  - ES-L1
  - GEO
  - HEO
  - SSO

Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/8f2521054a877aba23cb8470ca11098c74ff1479/2-2-jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

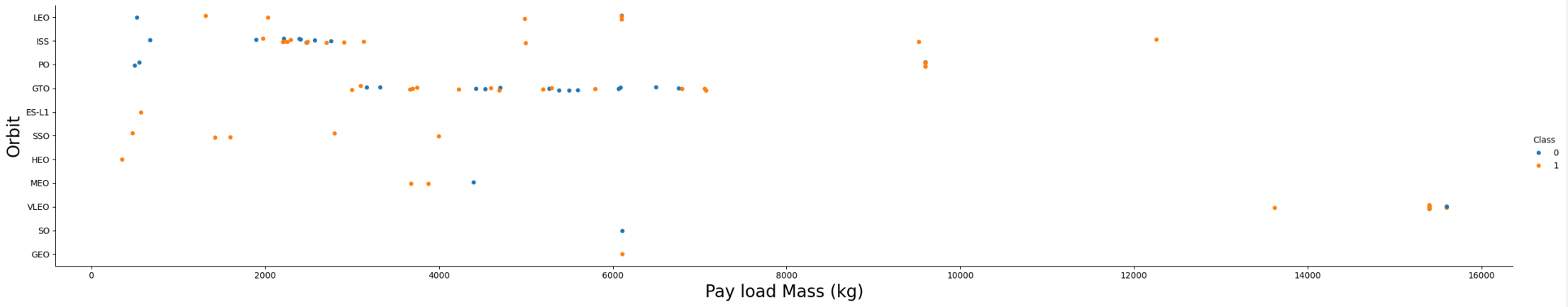
Flight Number vs. Orbit Type

- Plotted Flight Number vs. Orbit Type as a Scatter Plot:
  - In later flights additional Orbit Types are appearing, might be a new technical possibility?
  - Correlation between Orbit and Success seems to be indifferent:
    - In LEO orbit, success appears to be correlated to the number of flights
    - In GTO orbit, there seems to be no relationship between flight number and success.



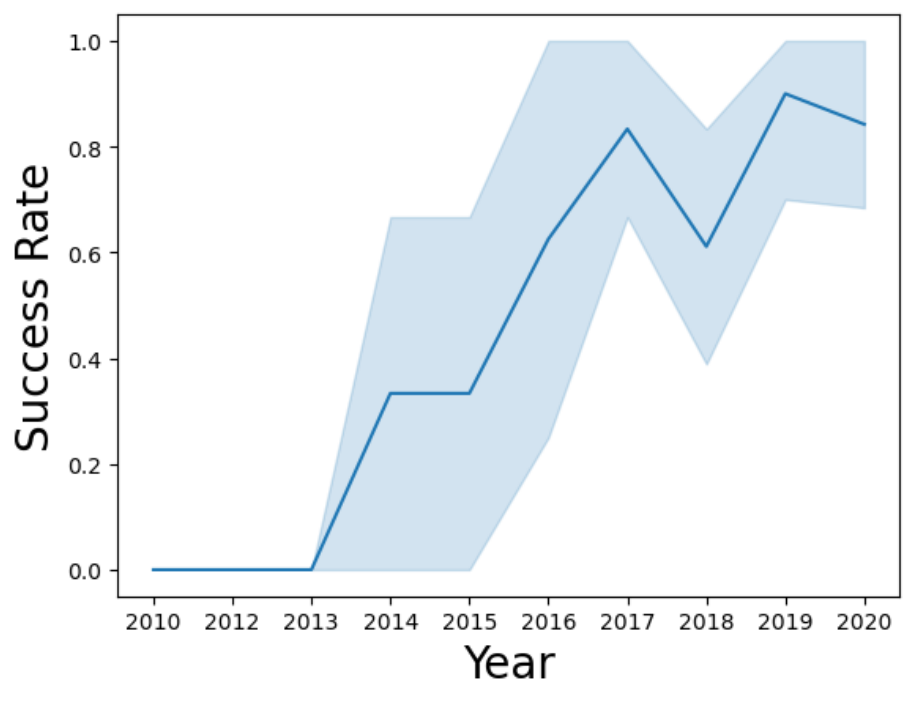
Payload vs. Orbit Type

- Plotted Payload vs. Orbit Type as a Scatter Plot:
  - ISS Orbit ends to have lower payloads
  - GTO Orbit tends to have moderate payloads
  - VLEO Orbit tends to have the highest payloads



Launch Success Yearly Trend

- Plotted Year of the launch vs the Success Rate for the years 2010 to 2020

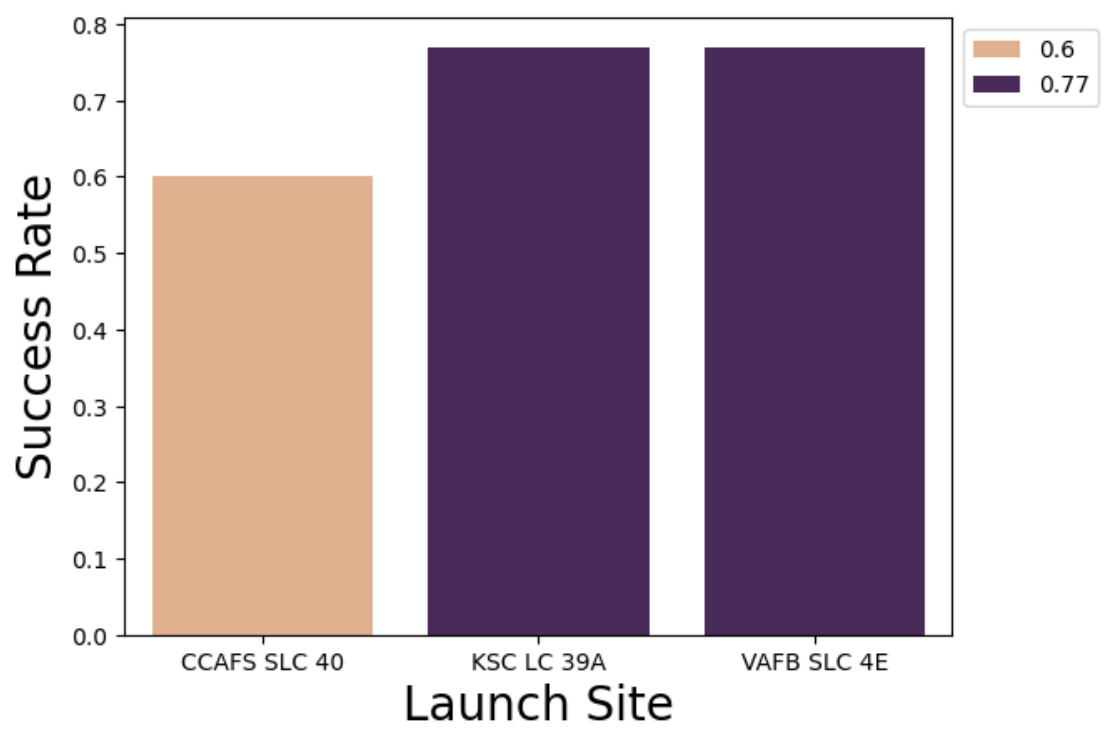


- correlation between the two variables seems to be strong
- This is likely to suggests
  - learnings from initial flights have been applied to later flights
  - technological, operational or other improvements have been made over time

Notebook @ GitHub:
https://github.com/danieljirausch/ibm-data-science/blob/8f2521054a877aba23cb8470ca11098c74ff1479/2-2-jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

Additional Findings - Success Rate vs. Launch Site

- We plotted a Bar Chart of Launch Site and Success Rate of Launches to see which Launch Sites where the most promising

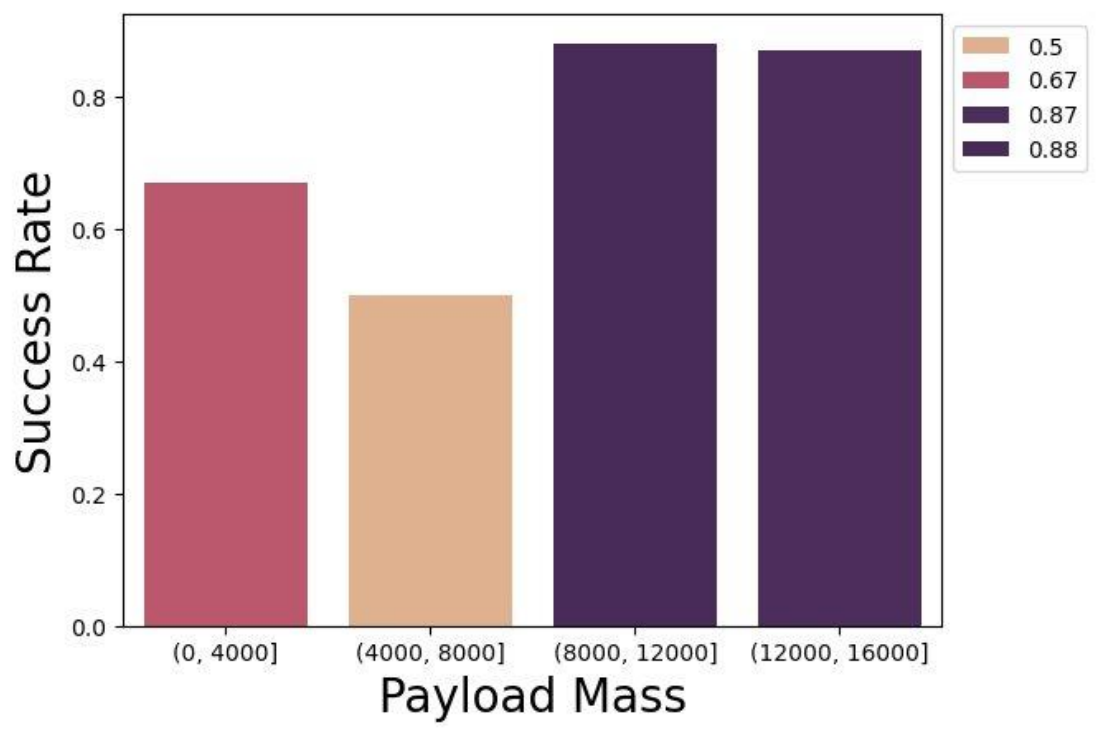


There might be an advantage to launch from

- KSC LC 39A
- VAFB SLC 4E

Additional Findings - Success Rate vs. Payload Bins

- We plotted a Bar Chart of Payload Mass (4 Bins 4000kg each) and Success Rate of Launches to see which are most successful



It appears that launches with a higher Payload have a higher success chance. The reasons for this should be evaluated further by cross referencing Payload mass with different other features.

All Launch Site Names

- Using **SELECT DISTINCT** we could identify four unique launch sites
- There are **four unique Launch Sites** in the dataset

Display the names of the unique launch sites in the space mission

```
%sql select distinct "Launch_Site" from SPACEXTBL;
```

```
 * sqlite:///my_data1.db
Done.
```

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Launch Site Names Begin with 'CCA'

- by filtering the SQL query using **WHERE** and **LIKE** with % Operator, we can extract Dataset with Launch Sites beginning with **'CCA'**
- The first five datasets are shown here

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- using **SUM,** we can determine the total value of a field
- for the Customer **Nasa (CRS)** the **PAYLOAD_MASS__KG_** is

Display the total payload mass carried by boosters launched by NASA (CRS)

```
%sql select sum(PAYLOAD_MASS__KG_) as sum_nasa from SPACEXTBL where "Customer" = 'NASA (CRS)';
```

```
 * sqlite:///my_data1.db
Done.
```

sum_nasa
45596

Average Payload Mass by F9 v1.1

- Similar to SUM, we can use **AVG** to calculate an average of a field
- The **average PAYLOAD_MASS__KG_** for the booster version "**F9 v1.1**" is

Display average payload mass carried by booster version F9 v1.1

```
%sql select avg(PAYLOAD_MASS__KG_) as avg_payload from SPACEXTBL where "Booster_Version" = 'F9 v1.1';
```

```
 * sqlite:///my_data1.db
Done.
```

avg_payload
2928.4

First Successful Ground Landing Date

- Using the **MIN** function, we can retrieve the minimum of a value
- The **first successful landing on a ground pad** was on

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
%sql select min(Date) as min_date from SPACEXTBL where "Landing_Outcome" = 'Success (ground pad)';
```

```
 * sqlite:///my_data1.db
Done.
```

min_date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Using **BETWEEN** in SQL enables us to filter a range
- These are the names of boosters which have **successfully landed on drone ship** and had **payload mass greater than 4000 but less than 6000**

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%sql select distinct "Booster_Version" from SPACEXTBL where "Landing_Outcome" = 'Success (drone ship)' and "PAYLOAD_MASS__KG_"
```

```
 * sqlite:///my_data1.db
Done.
```

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- Using **COUNT** enables us to get the number of entries for a specific field category
- There are four entries in the result, three match to successful
- Data should be cleaned after this finding

List the total number of successful and failure mission outcomes

```
%sql select "Mission_Outcome", count(*) as cnt from SPACEXTBL group by "Mission_Outcome";
```

```
 * sqlite:///my_data1.db
Done.
```

Mission_Outcome	cnt
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- There are a few different booster version that have carried the maximum load

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
%sql select "Booster_Version" from SPACEXTBL where "PAYLOAD_MASS__KG_" = (select max("PAYLOAD_MASS__KG_") from SPACEXTBL)
```

```
 * sqlite:///my_data1.db
Done.
```

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

2015 Launch Records

- We can extract the month from the Date column using **SUBSTR**
- There are only two launches with a **failed landing** in **2015**
- Both failed on a **drone ship**

```
%sql select substr(Date, 6,2) as month, "Landing_Outcome", "Booster_Version", "Launch_Site" from SPACEXTBL where substr("Date"
```

```
 * sqlite:///my_data1.db
Done.
```

month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

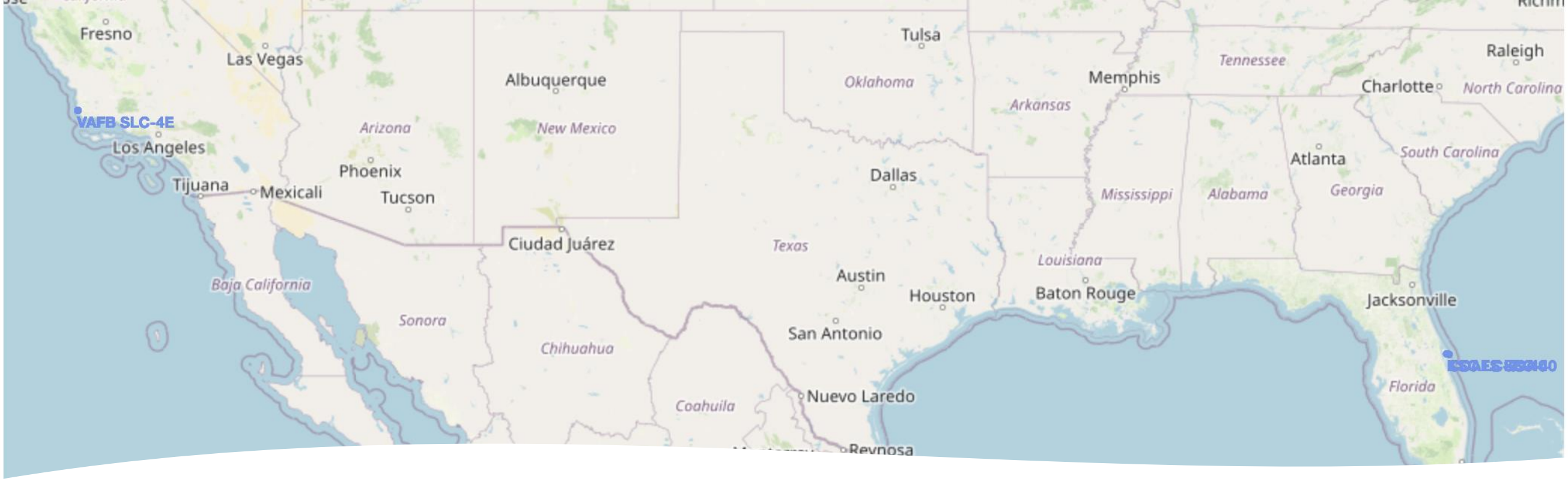
- COUNT and RANK can be used to rank the outcome of landing in the time frame
- **Most** landings where **not attempted**
- Second most common outcomes are **Success and Failure on a drone ship**

```
%sql select "Landing_Outcome", count(*), RANK() OVER (ORDER BY count(*) desc) cnt_rank from SPACEXTBL where "Date" between '20
```

```
 * sqlite:///my_data1.db
Done.
```

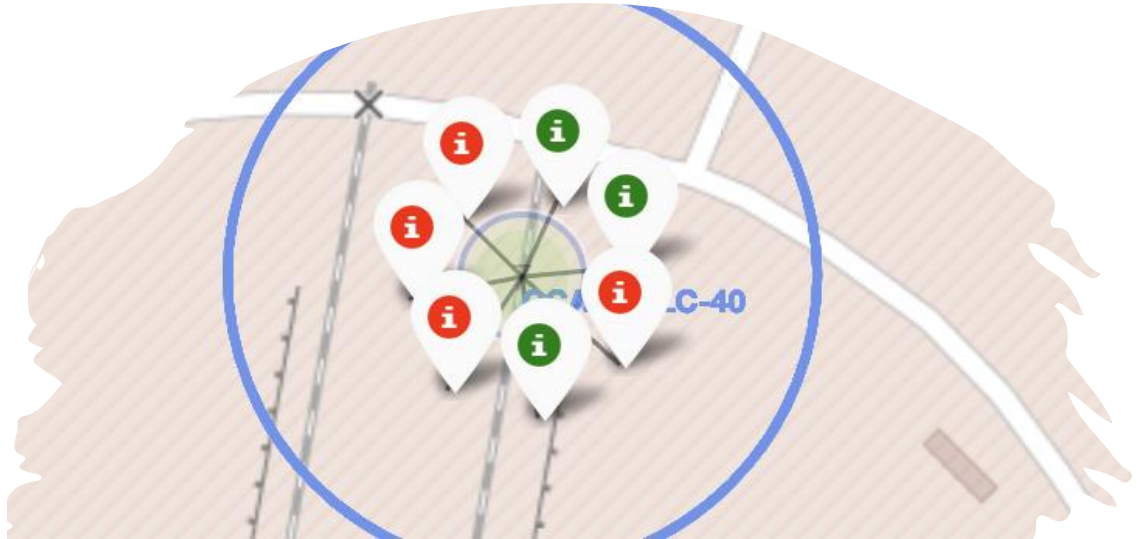
Landing_Outcome	count(*)	cnt_rank
No attempt	10	1
Success (drone ship)	5	2
Failure (drone ship)	5	2
Success (ground pad)	3	4
Controlled (ocean)	3	4
Uncontrolled (ocean)	2	6
Failure (parachute)	2	6
Precluded (drone ship)	1	8


Section 3
Launch Sites
Proximities Analysis



Launch Site Overview

- The map shows the locations of launch sites at a glance
- All launch sites are near the coast
- 3 of the 4 are in Florida, in close proximity

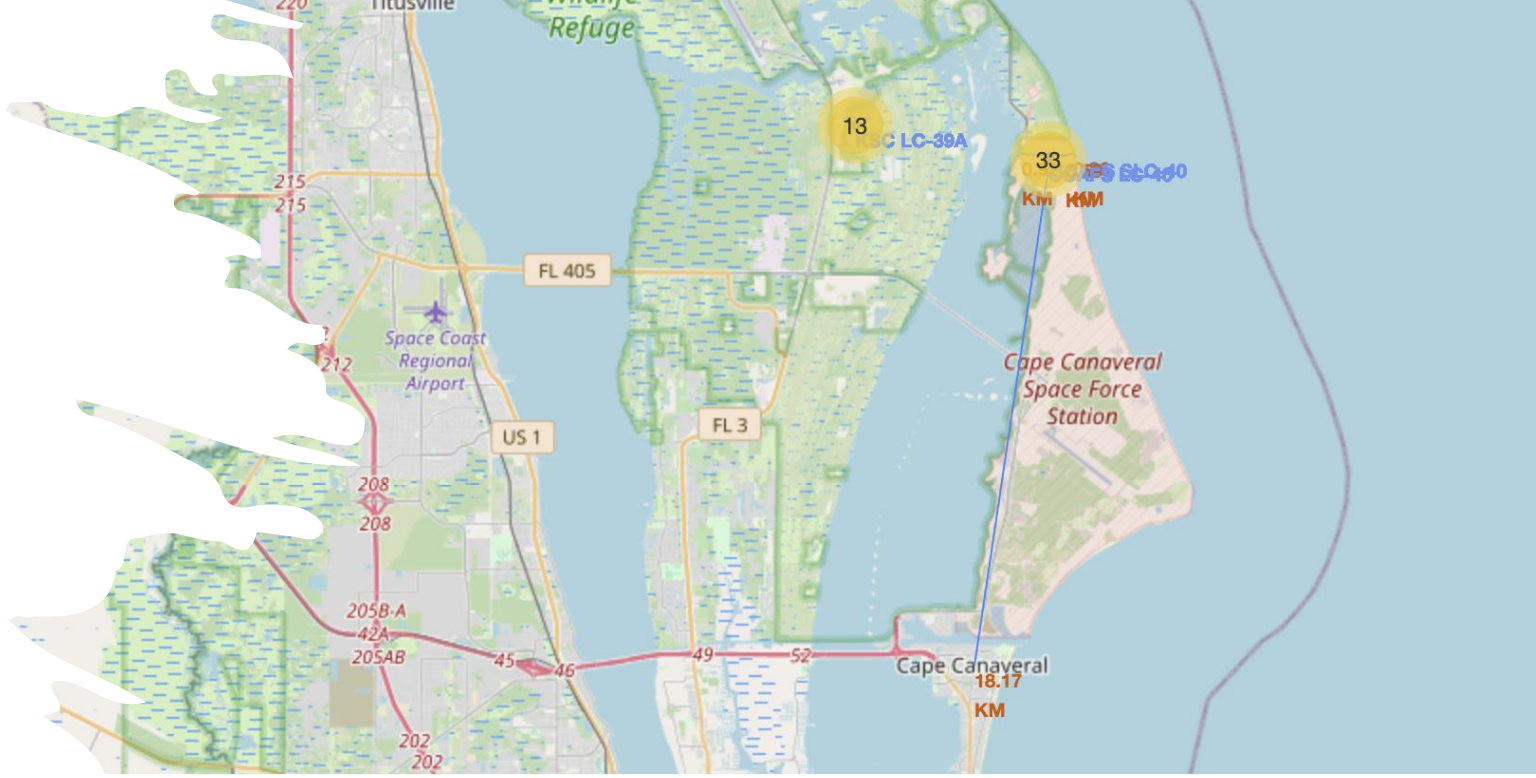


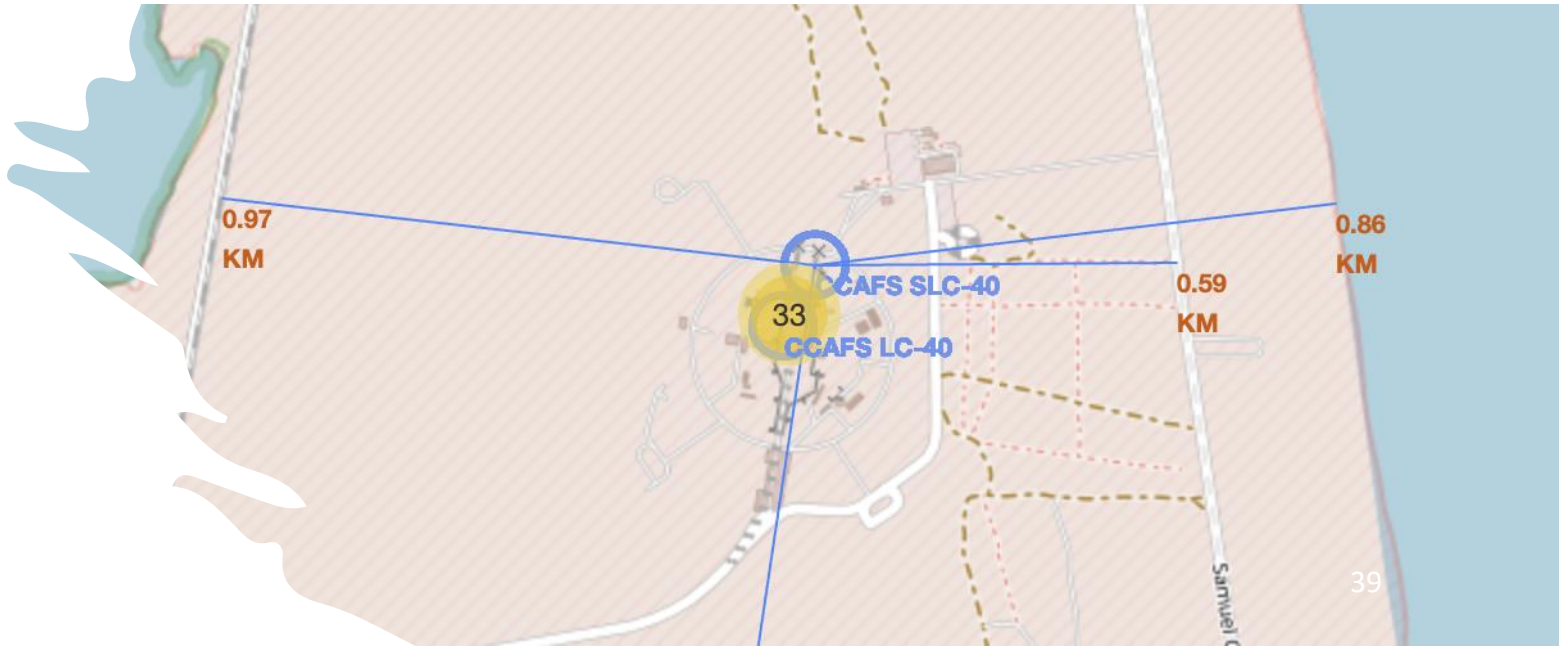
Launch Success for Site CCAFS SLC-40

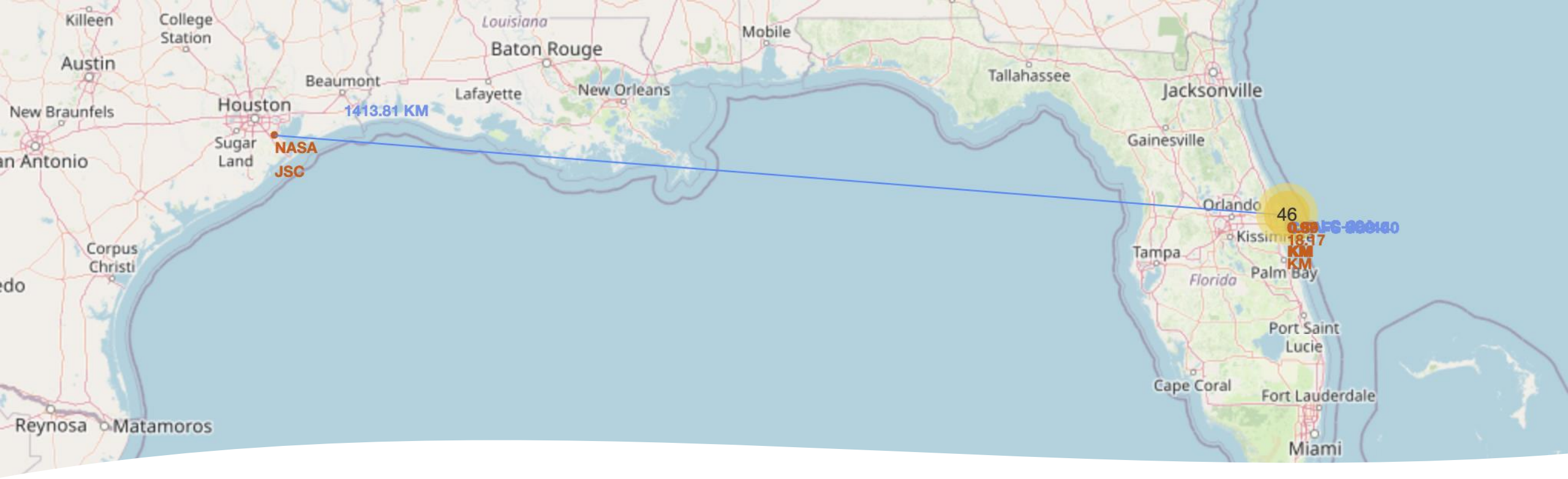
- Clicking on a launch site shows the landing success for that site
  - Red – failed landing
  - Green – successful landing
- For CCAFS SLC-40 success and failures are about even

Surroundings of CCAFS SLC-40 with distance

- The launch site has a close proximity to
  - Coast
  - Railroads
  - Highways
- The distance to towns is quite large



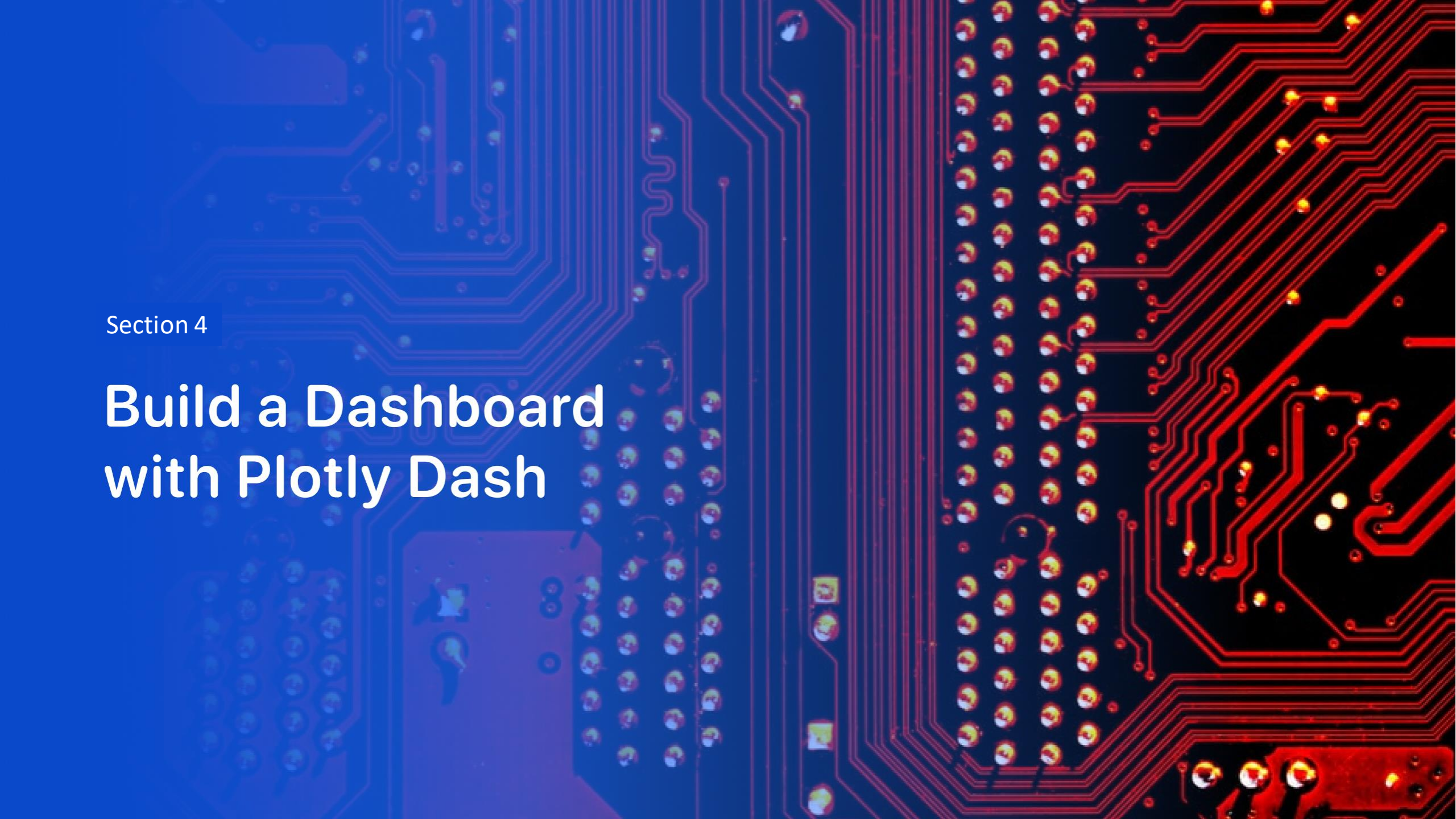




Additional Findings Distance to NASA

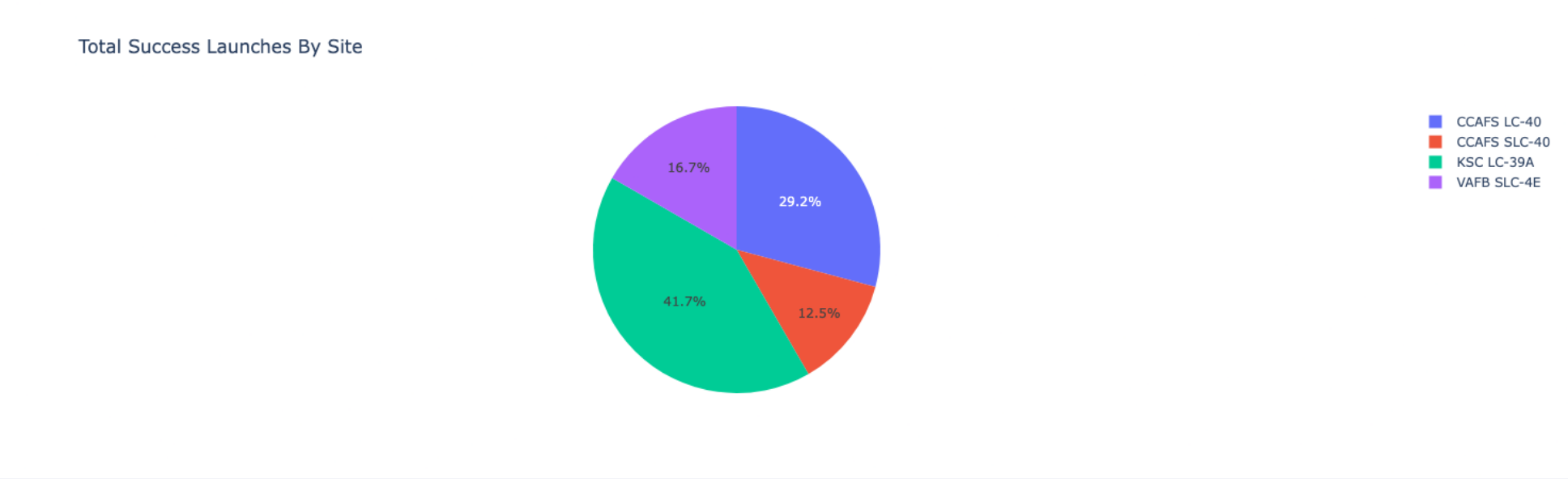
- The distance between launch site CCAFS SLC-40 and the NASA Headquarters in Houston is quite large

1413.81 km


Section 4
Build a Dashboard with Plotly Dash

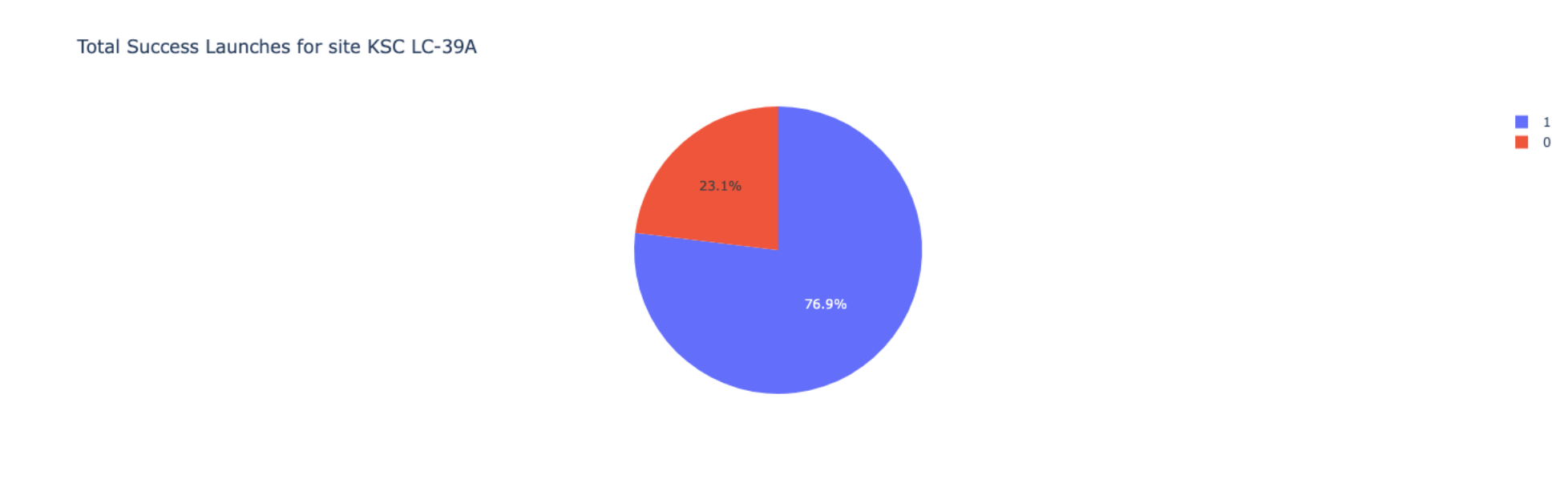
Total Success Launches By Site

- Launches were successful from all four sites
- The highest number of successful launches come from **KSC LC-39A** (41.7%)

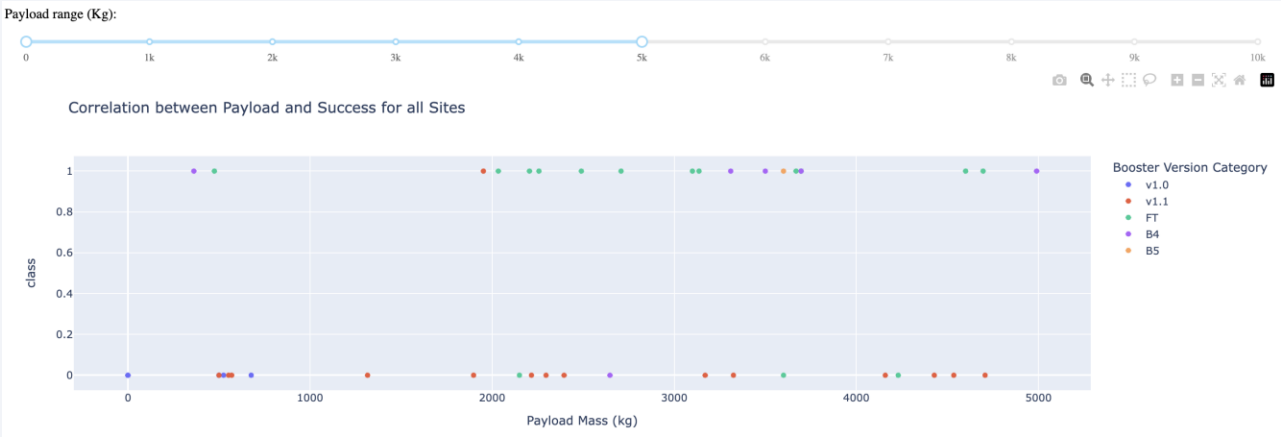


Launch success rate for KSC LC-39A

- More than **3 out of 4** launches from KSC LC-39A landed **successfully**
- The success rate is **76,9%**



Correlation between Payload and Success

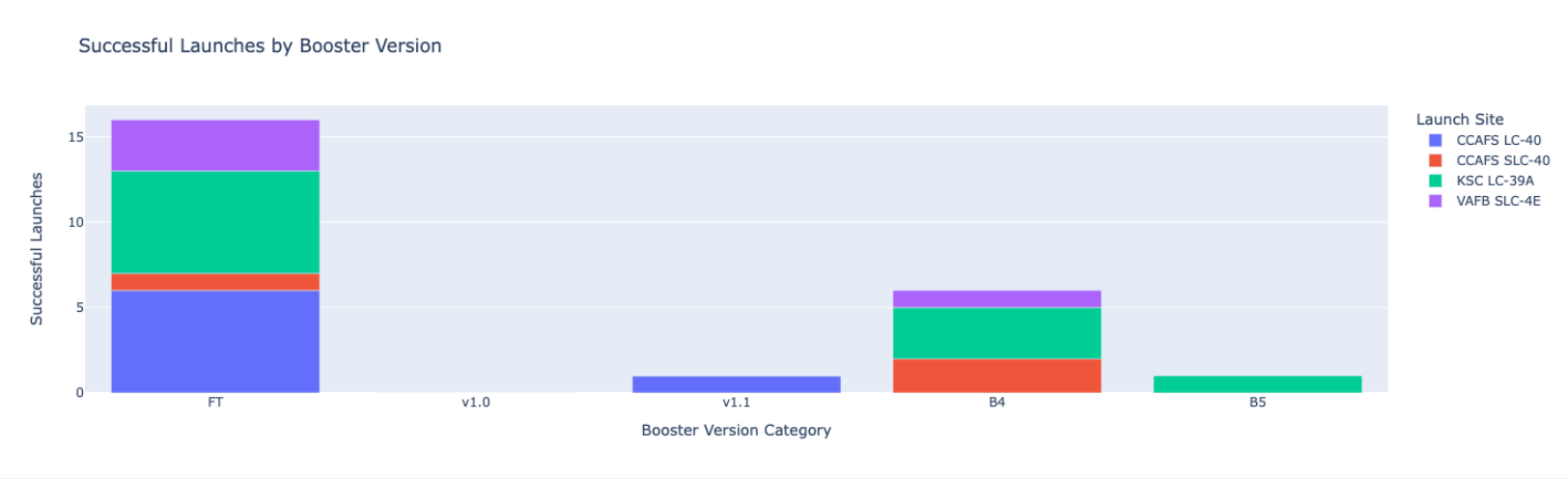


- For **payloads <5000** the distribution of failed and successful launches is almost **evenly split**
- The Booster Version **FT** has **high success rate** in that range
- Booster Version **v1.1** has a pretty **low success rate** in that range

- For **payloads >5000** there a **more failed** than successful launches
- There is **no significant difference** between Booster Version

Additional Findings – Successful launches by Booster Version

- By far the **most successful launches** where done using the **FT Booster Version**
- Booster **v1.0** had **no successful launches**
- These numbers seem to be **independent from the launch site**

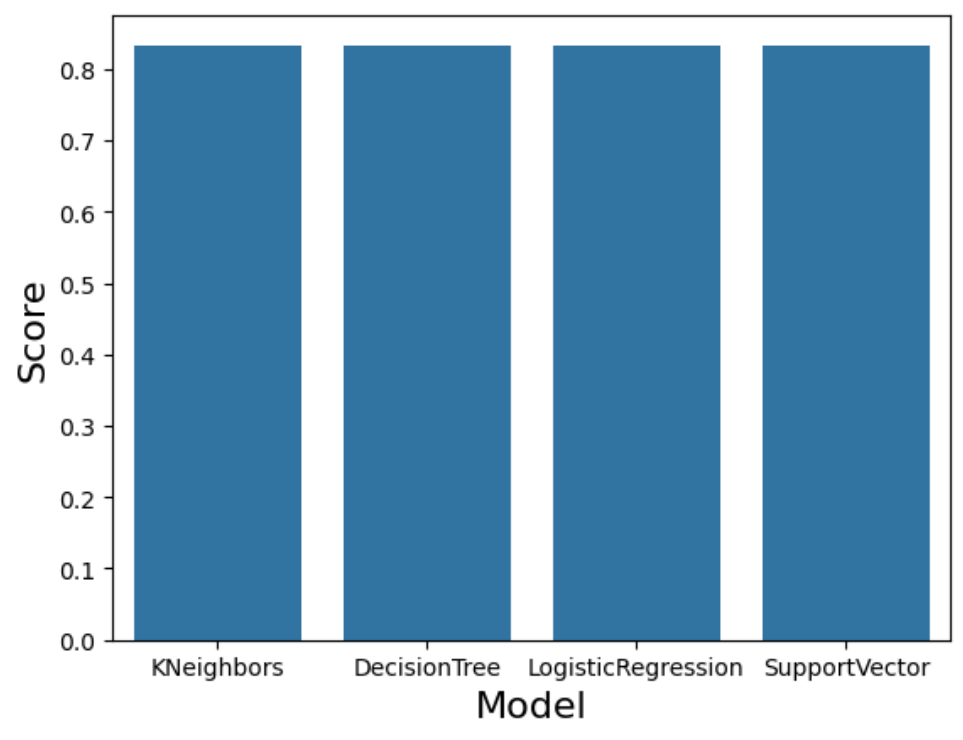


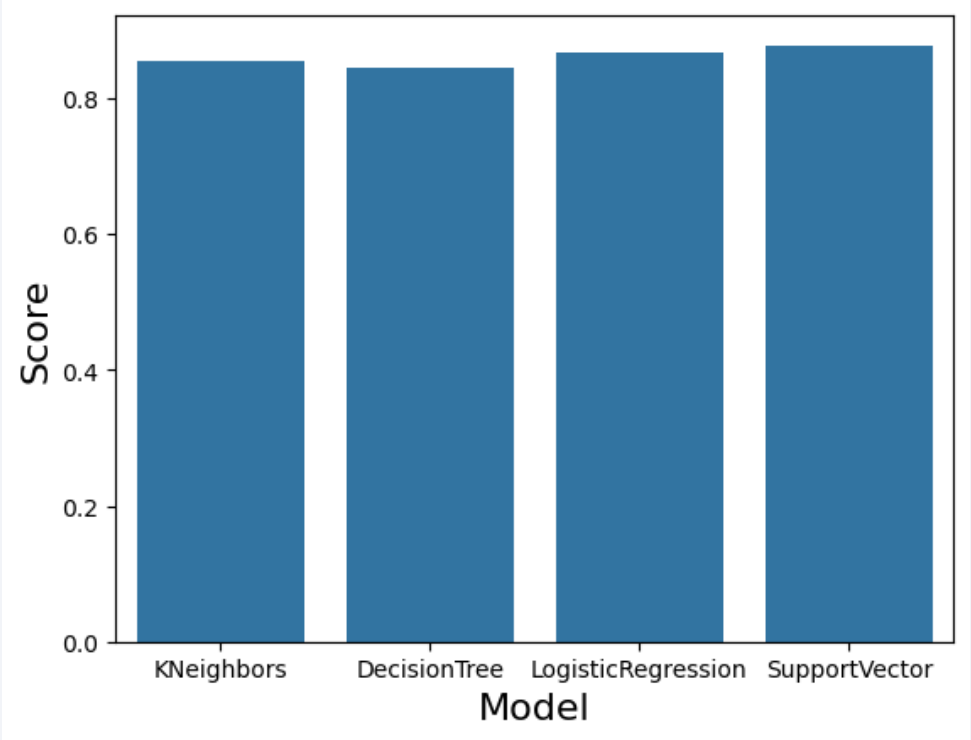
Section 5

Predictive Analysis (Classification)

Classification Accuracy

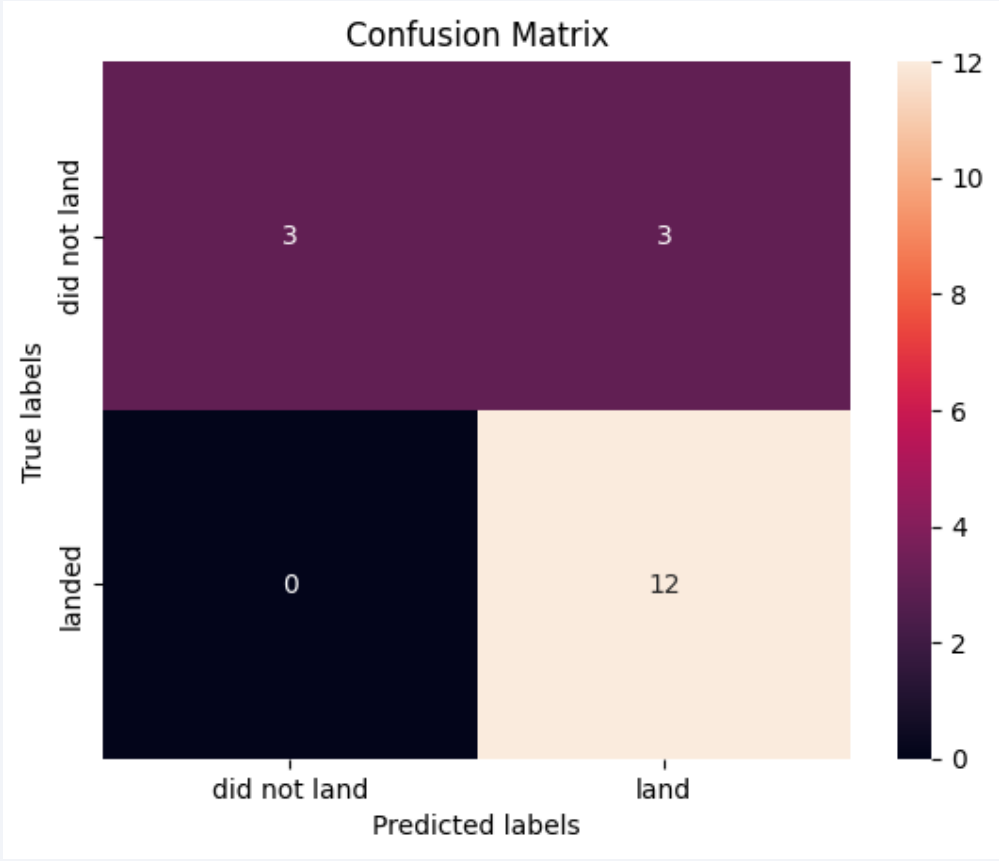
- With the current dataset, all models perform pretty equal regarding accuracy based on the test dataset
- Regarding accuracy against the whole data sample, Support Vector has the highest





Confusion Matrix

- The confusion matrix for all models looks similar:
  - Predicting a successful landing seems to work well as there are no false negatives
  - Predicting a failed landing yields as many false positives as correct predictions



Conclusions

- We can conclude that:
  - The launch success rate almost constantly increased from 2013 to 2020
  - Launch site KSC LC-39A had the most successful launches of any sites.
  - Orbits ES-L1, GEO, HEO, SSO, VLEO had the highest success rate.
  - A good launch site is determined by the proximity to a coast and transport infrastructure
  - Several classification models perform well in predicting successful launches

Additional findings

- Slide 24 - Success Rate vs. Launch Site
- Slide 25 - Success Rate vs. Payload Bins
- Slide 40 - Distance to NASA
- Slide 45 - Successful launches by Booster Version

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!